International Journal of
Science and Engineering Investigations
vol. 7, issue 75, April 2018
ISSN: 2251-8843

Recognized Objects Visualization on Maps

R. A. Simonyan
IIAP NAS RA
(simonyan.r2@gmail.com)

Abstract- At carrying out video surveillance in the area by different cameras, each camera detects its object, which is considered as a separate unit [1, 2]. Thus, the whole image of the space is not obtained and remains unknown. But there are tasks for whose solution, it is necessary to see a complete picture considering the interrelatedness of the objects detected by different video cameras.

It becomes necessary to combine the objects detected by all video-cameras in one system, determine the coordinates of the latter and the position of the objects in relation to each other analyze and make calculations in the unified system [4, 5].

This article describes the selected method and the software implementation for the solution of the mentioned task.

Keywords- *Visualization of Objects on Maps, Cartographic Tool for Visualization of Objects, Recognition of Objects During Video Surveillance, Open Source Application for Drawing Objects, Layout Creation*

I. Introduction

The purpose of the program is to demonstrate the combined objects (monitoring) on a single platform (for example, a topological map [7]), as well as to preserve the trajectory and history of the movement of objects in the previous days for their study and analysis in the future. There is also a simultaneous demonstration of objects detected by video cameras on the cartographic platform [5, 6], i.e. monitoring of the live broadcast.

To solve the problem it is necessary to develop a method that should have the following functions:

1. Displaying the objects by the coordinates on one topological platform–map;

2. Monitoring of the live broadcast;

3. Saving the location of objects on the map as a project;

4. Loading of the previously saved projects;

5. Making changes in the project (move objects, erase, add a new one, zoom, etc.)

II. Description of the System

The field of general cartography, as well as tools for developing a cartographic editor (including Java Script, HTML, and CSS description languages) have been studied [7-11].

As already noted, the system should have a screen to display objects, buttons to control actions on the screen, as well as the ability to save and load the created project.

Let’s discus the structure and capabilities of the system. The system consists of 3 parts: internal view, collection of functions, appearance (Fig. 1).

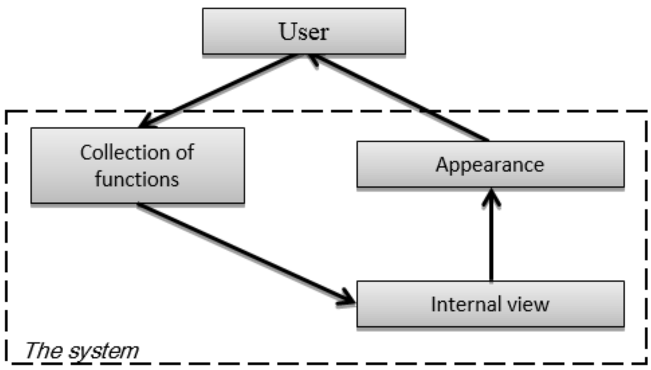

Figure 1. The general structure of the system

The system works as follows: the User sees the “Appearance” of the system, then, to perform the actions, refers to the “Collection of functions”. The latter, in turn, establishes a connection with the ”Internal view“, which, after the actions, displays the result in the ”External view". The “Internal view” and the “Collection of functions” are the code of the whole program.

Actions are carried out by buttons, the screen shows the process of all actions, and their results. Commands have been developed in the software tool. Each command is represented by a separate algorithm. The primary command is to draw/add an object by coordinates.

Numerous commands have been developed to introduce changes in the project, for example, to add additional objects for the purpose of their consideration, calculation and analysis in relation to the present object. Among the additional commands are: add new object, move, stretch/transform, copy, delete, select, rotate, add text, place grid, grid size change, ruler, delete all from the screen.

To add more objects on the screen, in the system there are standard samples of objects which can be added or withdrawn within the software tool.

Let us consider the function of displaying objects on the screen.

The command to add an object works as follows: after selecting an object, at the moment of drawing the object (the new object is not yet displayed on the screen), the following parameters are fixed in the code: the x, y coordinates, width, length, and colour. Using these parameters, an object is formed and registered in the array where the objects drawn on the screen are already stored (Fig. 2).

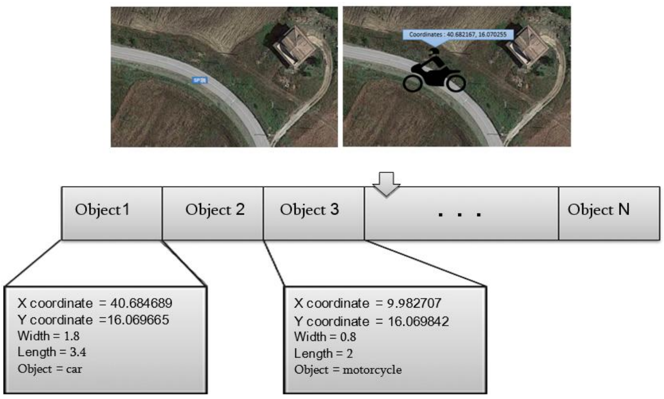

Figure 2. Software representation of adding an object on the screen

The following operation takes place instantly: objects are deleted from the screen and the function of adding objects reads by cycle all the elements of the array and draws them on the screen.

Since the system allows the user to perform actions on the objects, we will discuss some of them.

Basic commands include moving and copying of the object.

The algorithm developed for moving the object is as follows: select an object while the mouse is in the pressed state and the cursor is in motion, the object is deleted from the screen and a new object is drawn by the cursor coordinates (Fig. 3). This happens continuously, with very small intervals (about 20m/sec) while the cursor is in motion. The value of this time period is arbitrary, but should be very short, so that the absence of the object should not be noticed. As a result, the object moves on the screen (Fig. 4).

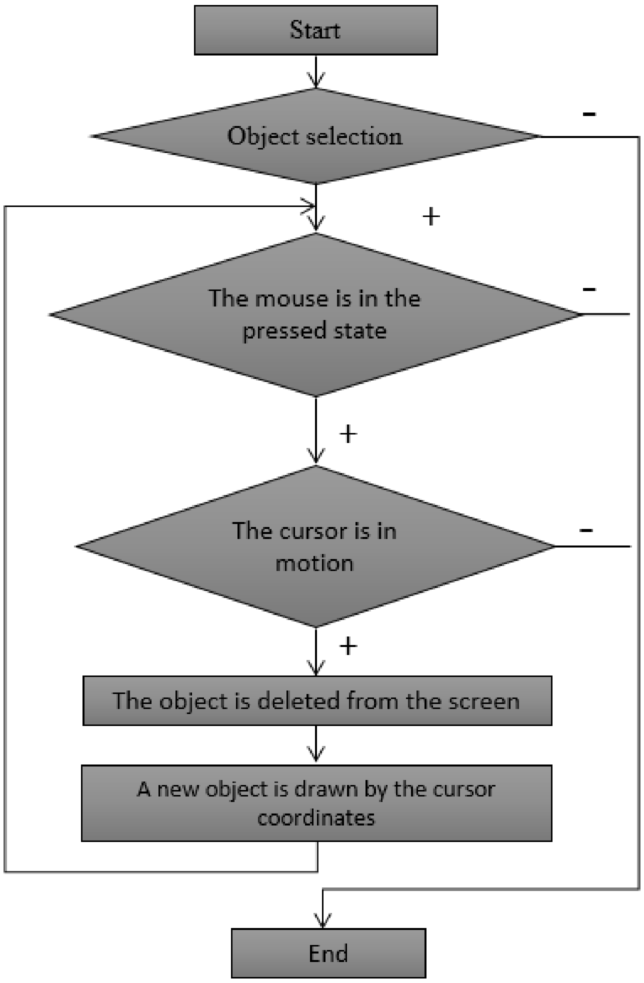

Figure 3. Algorithmic block-diagram of the functional block of object movement

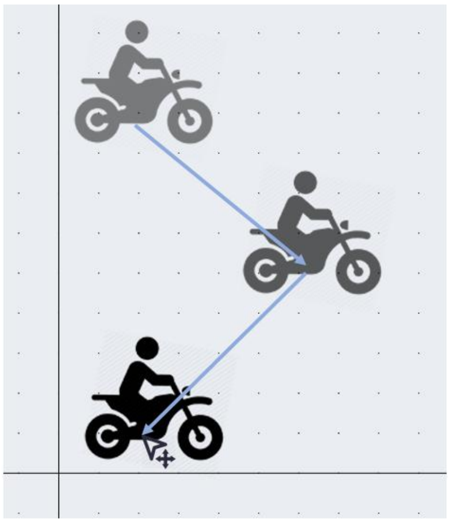

Figure 4. The object movement on the screen

When copying an object, a copy of the selected object (using the add function) is created which does not differ from the previous one. When you call the copy function, the move function is also activated, with which the copy can be moved. On the example given in Fig. 5, the copied object is moved from the position of the cursor 1 to 2.

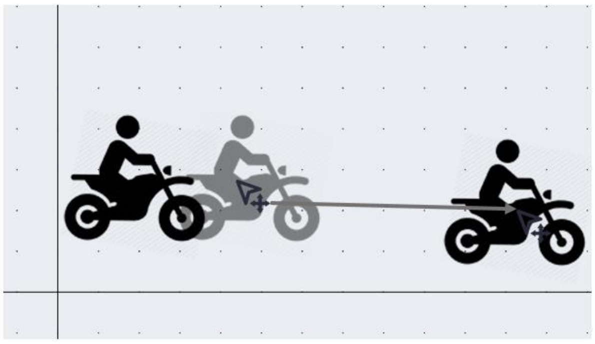

Figure 5. The copied object and its movement

III. THE INTERFACE STRUCTURE OF THE SOFTWARE TOOL

In the structure, there are buttons and a screen (Fig. 6). The map is placed on the screen. The screen displays objects by coordinates. Using the buttons, , new objects are added on the screen and actions are performed with those objects. The screen shows all actions and their results.

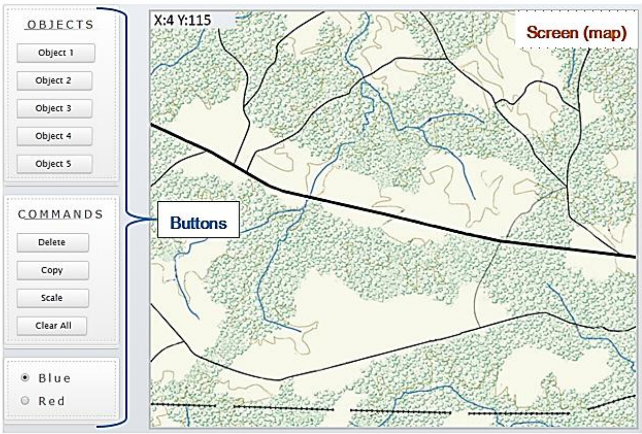

Figure 6. The interface structure

The editor is intended for carrying out cartographic actions for different purposes. Let us consider several types:

* Cartography of geographical area (Fig. 7);

* Cartography of the city (Fig. 8 and 9);

* Cartography of the area of the plant or other special-purpose territory (Fig. 10);

* Animated cartography (Fig. 11);

* An example of general cartography (Fig. 12).

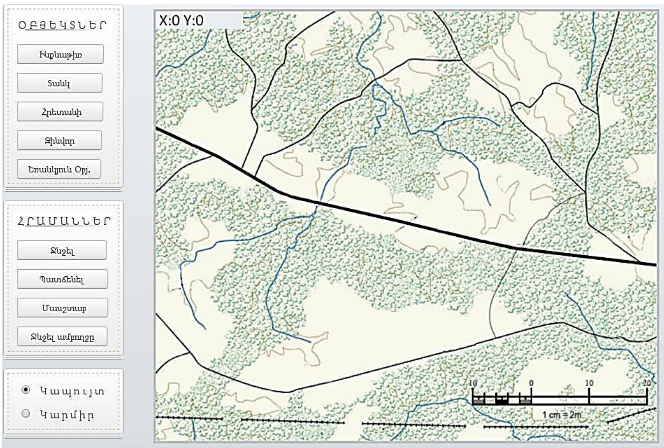

Figure 7. Cartography of geographical area

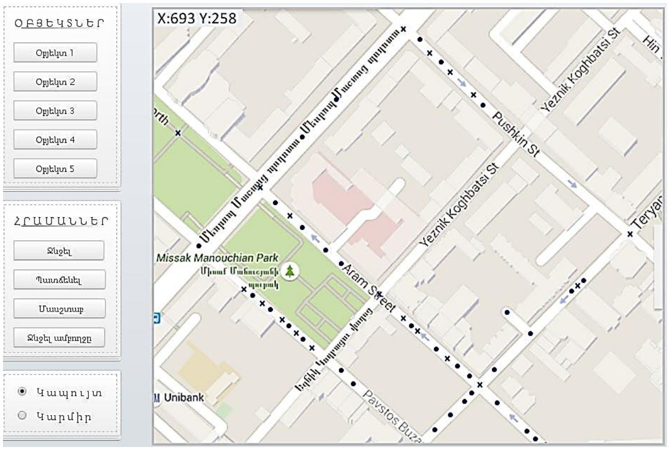

Figure 8. Cartography of the city

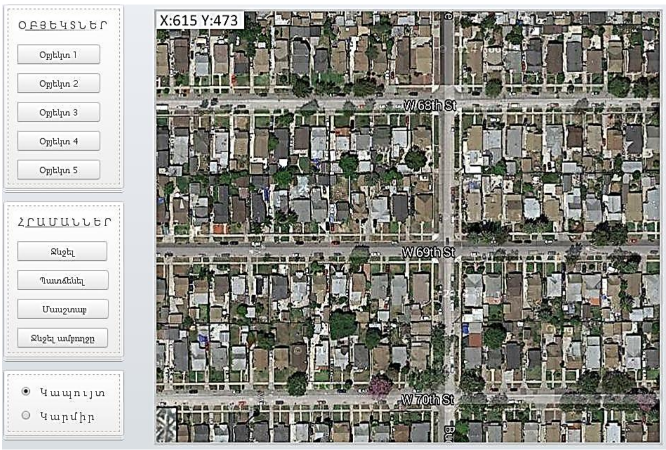

Figure 9. Cartography of the city

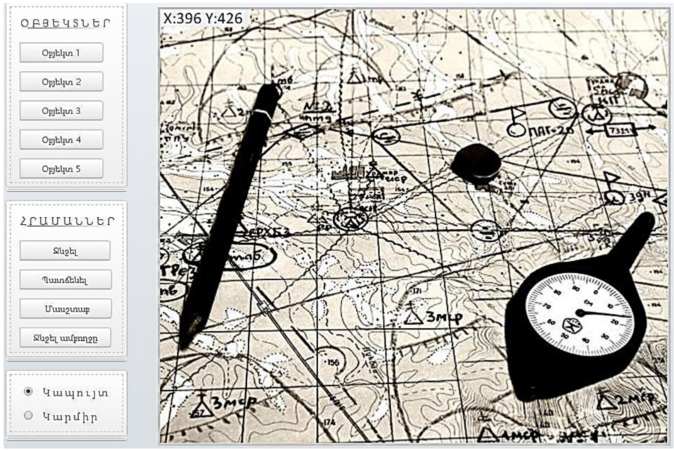

Figure 12. An example of general cartography

Figure 10. Cartography of the area of the plant or other special-purpose territory

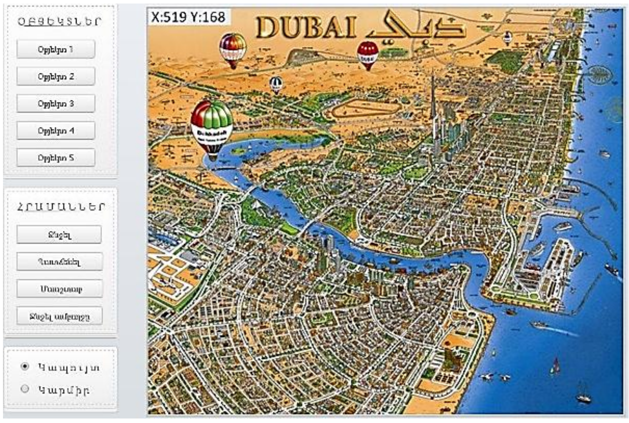

Figure 11. . Animated cartography

Within the software tool, a function for saving and loading is developed. MS SQL database and Python programming language have been used for the software implementation of the latter. To save the project, it is necessary to enter the Project Name, then, by clicking on the button "Save", information of the graphical models is sent to the server, by frequent inquiries, where it is entered into the database. the already saved project can be loaded from the database. This action is also carried out by frequent inquiries, after which all the information is taken from the database of the project and sent back to the client, where it is displayed on the screen through the browser. An intermediate stage has been developed, allowing the user to see the project in a text format, change it, and then load it on the screen before the project is shown on the screen.

IV. Conclusion

Thus, a method has been developed, allowing to combine the objects detected by all cameras into one system, i.e. to display objects in one topological platform – map by coordinates, to monitor the live broadcast, to save the position of objects on the map as a project, to load the projects saved earlier and to introduce changes in the project (move objects, delete, add a new one, zoom, etc.).

The developed software tool is an open source. Thanks to this, it can be easily changed.

The system can be used to solve various topological problems (cartography, topological design of chips).

REFERENCES

[1] R. A. Simonyan, “Hidden and Unknown Object Detection in Video”, In International Journal of New Technology and Research (IJNTR) Volume 2, Issue 11, pp. 22-25, November 2016.

[2] R. A. Simonyan, D. A. Simonyan, “Detection and ignorance method of false targets during object detection”, In Computer Science and Information Technologies (CSIT) Conference 2017, pp. 372-375

[3] R. A. Simonyan, Web application for physical design of microsystems, “Lraber Scientific Articles Collection”, Annual Conference of NPUA, 2015

[4] P. Muukkonen and 1. Heiskanen, "Biomass Estimation Over a Large Area Based on Standwise Forest Inventory Data and ASTER and MODIS Satellite Data: A Possibility to Verify Carbon Inventories," Remote Sensing of Environment, 107, 4, pp.617-624, 2007.

[5] R. Collins, A. Lipton, and T. Kanade, “Introduction to the special section on video surveillance,” IEEE Trans. Pattern Anal. Machine Intell., vol. 22, no. 8, pp. 745–746, August 2000.

[6] H. D. V. Bohmand F. Siegert, "Remote Sensing Verification by Aerial Surveys and Ground Truth Campaigns 1997 and 1998 in Central Kalimantan, Indonesia-Peat Swamp Forest, Mega-Rice-Project and Fires," Workshop on Tropical Forest and Remote Sensing, 1999.

[7] Nogami, D. Phuon, and M. Kusanagi, "Field Observation Using Flying Platforms for Remote Sensing Education, "Asian Conference of Remote Sensing, 2002.

[8] Cosentino V., Cánovas Izquierdo J. L., Cabot J., “A Systematic Mapping Study of Software Development With GitHub”, In IEEE Open Access Journal, Vol. 5, pp. 7173 – 7192, March 2017.

[9] Fett D., Küsters R., Schmitz G., “An Expressive Model for the Web Infrastructure: Definition and Application to the Browser ID SSO System”, In IEEE Symposium on Security and Privacy, May 2014.

[10] Babovic Zoran B. , Protic J., Milutinovic V., “Web Performance Evaluation for Internet of Things Applications”, In IEEE Open Access Journal, Vol. 4, pp. 6974 – 6992, October 2016.

[11] Ofuonye E., Miller J., “Resolving JavaScript Vulnerabilities in the Browser Runtime”, In 2008 19th International Symposium on Software Reliability Engineering (ISSRE), Nov. 2008.